

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

BAKALÁŘSKÁ PRÁCE



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

DEPARTMENT OF CONTROL AND INSTRUMENTATION

ŘÍZENÍ DOPRAVNÍKOVÉHO SYSTÉMU PRO ZKUŠEBNU

CONTROL OF CONVEYOR BELT FOR TESTING ROOM

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

Michal Lorenc

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Tomáš Macho, Ph.D.

BRNO 2020

Bakalářská práce

bakalářský studijní program **Automatizační a měřicí technika**

Ústav automatizace a měřicí techniky

Student: Michal Lorenc

ID: 203281

Ročník: 3

Akademický rok: 2019/20

NÁZEV TÉMATU:

Řízení dopravníkového systému pro zkušebnu

POKYNY PRO VYPRACOVÁNÍ:

1. Seznamte se s požadavky na fungování dopravníkového systému pro zkušebnu elektromotorů. Definujte požadavky na řízení dopravníkového systému a požadavky na ošetření poruchových stavů.
2. Seznamte se s řídicím PLC, vývojovým prostředím Tia portál, použitými čidly a akčními členy a ovládacími prvky.
3. Navrhněte koncepci řídicího software pro řízení linky.
4. Vytvořte řídicí program pro PLC v prostředí Tia portál.
5. Program odlaďte na dopravníkovém systému ve zkušebně elektromotorů ve firmě SIEMENS elektromotory s.r.o., Mohelnice.
5. Popište dosažené výsledky a navrhněte případná další zlepšení řízení dopravníkového systému.

DOPORUČENÁ LITERATURA:

Jirgl, M. Laboratorní cvičení BPGA - Cvičení se systémy SIEMENS. FEKT VUT v Brně.

Termín zadání: 3.2.2020

Termín odevzdání: 8.6.2020

Vedoucí práce: Ing. Tomáš Macho, Ph.D.

doc. Ing. Václav Jirsík, CSc.
předseda rady studijního programu

UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRAKT

Práce se zabývá programováním dopravníkového systému, ovládaného pomocí PLC. První částí je rozbor řízené technologie a popis požadované funkce. Následuje seznámení s programovacím prostředím, PLC, čidly, akčními členy a ovládacími prvky. V druhé části je návrh řešení funkce dopravníkového systému a jeho implementace. Poté je celý systém odzkoušen, zhodnocen a jsou podány návrhy na zdokonalení.

KLÍČOVÁ SLOVA

Programovatelný logický automat, řízení, dopravníkový systém, dopravník, Tia portal, čidlo, akční člen, ovládací prvek.

ABSTRACT

The work deals with programing of conveyor system, controlled by PLC. The first part is analysis of the controlled technology a describition of required function. The following is an introduction to the programming environment, PLC, sensors, actuators and controls. Second part is a proposal for the solution of the function of the conveyor system and its implementation. Then the whole system is tested, evaluated and suggestions for improvement are submitted.

KEYWORDS

Programmable logic controler, control, coveyor system, conveyor, Tia portal, sensor, actuator, control.

LORENC, Michal. *Řízení dopravníkového systému pro zkušebnu*. Brno, 2020, 47 s. Bakalářská práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav automatizace a měřicí techniky. Vedoucí práce: Ing. Tomáš Macho, Ph.D.

PROHLÁŠENÍ

Prohlašuji, že svou bakalářskou práci na téma „Řízení dopravníkového systému pro zkušebnu“ jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno 7.6. 2020

.....
podpis autora

PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu diplomové práce panu Ing. Tomáši Machovi, Ph.D. za odborné vedení, konzultace a podnětné návrhy k práci. Děkuji také Ing. Ondřeji Brotanovi a panu Liborovi Dočekalovi za užitečné rady při praktické tvorbě práce.

Brno 7.6. 2020

.....

podpis autora

Obsah

1	Úvod	9
2	Popis řízené technologie	10
2.1	Komponenty dopravníkového systému	10
2.2	Požadavky zákazníka na řízení dopravníkového systému	12
2.3	Umístění čidel, akčních členů a ovládacích prvků	14
3	Programovatelný automat	15
3.1	PLC Siemens	15
3.1.1	Siemens CPU 1214C DC/DC/DC	16
3.1.2	Siemens SM 1221	16
3.1.3	Siemens SM 1222	16
4	Vývojové prostředí Tia Portal	18
4.1	Programovací jazyky	19
4.2	Programovací bloky	20
4.3	WinCC	20
5	Komponenty řízené technologie	21
5.1	Čidla	21
5.2	Akční členy	22
5.3	Ovládací prvky	23
6	Software	24
6.1	Návrh koncepce řídicího programu	24
6.2	Implementace	27
6.2.1	Stav linky	28
6.2.2	Signalizace	29
6.2.3	Přejezd	30
6.2.4	Automatický chod	33
6.3	Modifikace dopravníkového systému	40
7	Odladění a zhodnocení dopravníkového systému	42
7.1	Odladění dopravníkového systému	42
7.2	Zhodnocení a návrhy na zdokonalení	43
8	Závěr	44
	Literatura	45

Seznam obrázků

2.1	Dopravníkový systém-náčrtek	10
2.2	Dopravníkový systém-model	11
2.3	Dopravníkový systém-náčrt funkce	12
2.4	Dopravníkový systém-umístění čidel a ovladačů	14
3.1	Fotka PLC v rozvaděči	15
3.2	SM 1221 [7]	16
3.3	SM 1222 [8]	16
3.4	Tabulka CPU 1214C DC/DC/DC	17
3.5	Tabulka Siemens SM 1221	17
3.6	Tabulka Siemens SM 1222	17
4.1	Prostředí Tia Portal	18
4.2	Ikony jednotlivých bloků	20
5.1	Čidlo Sick [13]	21
5.2	Čidlo Balluf [14]	21
5.3	Čidlo festo [16]	21
5.4	Elektromotor [11]	22
5.5	Pneumatický píst [18]	22
5.6	Čidlo Pepperel [17]	22
5.7	Fotografie části rozvaděče	23
5.8	Fotografie ovladače na stanici kontroly	23
6.1	Diagram pro přejezd mezi dopravníky	25
6.2	Diagram pro přívoz palety na D1	25
6.3	Diagram pro odvoz palety z D10	26
6.4	Blok stav linky - program	28
6.5	Signalizační sloupek [12]	29
6.6	FB přejezd A	30
6.7	FB přejezd B	31
6.8	Blok přejezdu B - program	32
6.9	Funkční blok přejezd B - vnitřní proměnné	33
6.10	Diagram průběhu inicializace	34
6.11	Konkrétní instance bloku pro přejezd mezi D4 a D5	34
6.12	Zajištění vozíku na začátku linky - program	35
6.13	Zajištění vozíku na konci linky - program	36
6.14	Kontrola poruchy - program	37
6.15	Motory - program	38
6.16	Blokace na koncích linky - program	39

1 Úvod

S postupem času, kdy lidstvo shromažďuje více a více znalostí, dochází k pokrokům v technologiích. Existuje široká škála směrů, ve kterých dochází k pokrokům. Tato práce je zaměřena na odvětví průmyslové automatizace. To je jedno z lidem bližších aniž by si to uvědomovali. Protože například hromadná výroba nějakého výrobku ovlivňuje jeho cenu a dostupnost.

Práce se tedy zabývá oblastí průmyslové automatizace, zde konkrétně dopravníkovým systémem pro zkušebnu elektromotorů. Protože se jedná o zakázku pro jistou firmu, tak se práce občasné odkazuje na požadavky klienta a to ve směru konstrukce, funkčnosti nebo značky použitých zařízení.

Na začátku práce bude uveden technologický popis, kde bude vysvětleno z jakých částí se celý systém skládá. Dále jakou přesně má vykonávat funkci. Krátce bude definováno co je to PLC a trochu podrobněji bude rozebrán konkrétní použitý typ. V návaznosti na použité PLC bude seznámení se s prostředím pro jeho programování, neboli Tia portal. Poté budou rozebrány použité akční členy, ovládací prvky a čidla.

Následně bude vytvořen návrh na řídicí program a později bude implementován. Po dokončení programu se práce bude věnovat jeho odladění a uvedení do provozu na dopravníkovém systému. Zde budou řešeny problémy, které vznikly během programování bez odzkoušení, problémy plynoucí z praktického používání budou řešeny případnou modifikací programu. Celý dopravníkový systém bude zhodnocen a případně budou navrženy změny pro zdokonalení.

2 Popis řízené technologie

2.1 Komponenty dopravníkového systému

Dopravníkový systém pro zkušebnu (dále také uváděn jako linka), název mluví sám za sebe. Jedná se o dopravníkový systém, který transportuje palety na místo určení. Dopravník je válečkový s pohonem. Pro lepší popis lze dopravníkový systém rozdělit na vozík pro přívoz a odvoz palet. Dále na číslované dopravníky D1 až D10 a stanice přívozu, kontroly, vyřazení a odvozu. Případně, ale to je nepodstatné dělení, na část uvnitř a vně kontrolní kabiny.

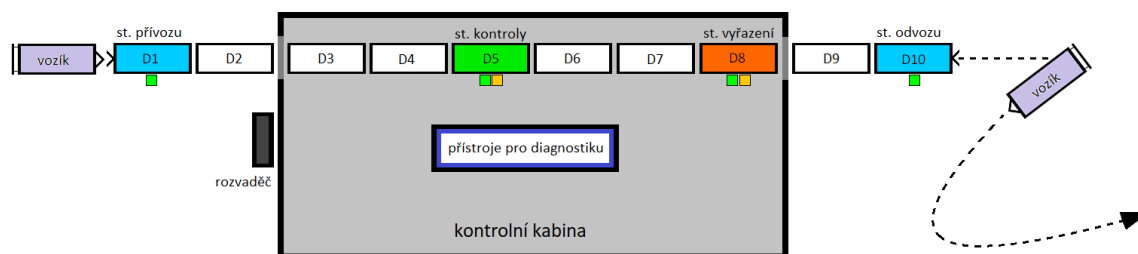
Vozíky

Jsou umístěny resp. lze je připojit na začátek resp. konec linky, proto vozík pro přívoz palet a druhý pro odvoz palet. Můžeme je vidět na obrázku 2.2. První vozík je připraven k připojení na začátek dopravníkového systému. Druhý vozík odváží paletu.

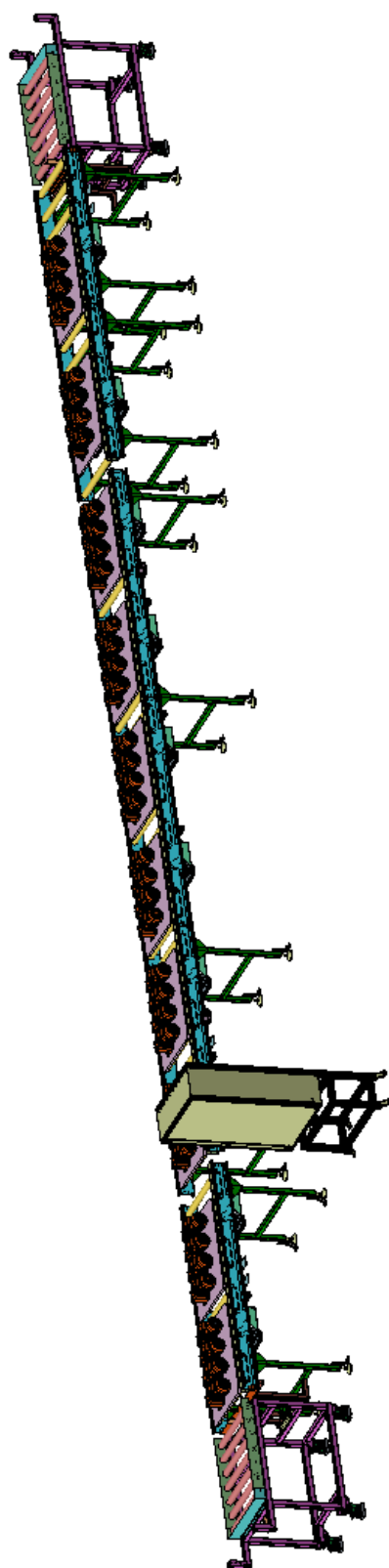
Jednotlivé dopravníky

Celá soustava dopravníků může být rozložena na 10 malých dopravníků, které jsou dále v práci označovány zkráceně písmenem D a číslem (např. D1). Dopravníky číslo D1, D2, D9, D10 jsou mimo kontrolní kabinu, ostatní dopravníky jsou uvnitř kabiny. Vše je znázorněno na obrázku 2.2.

- **Stanice přívozu**- z vozíku se paleta přesouvá na začátek dopravníku.
- **Stanice kontroly**- zde je provedena kontrola motoru resp. motorů (diagnostika).
- **Stanice vyřazení**- je místo k vyřazení nevyhovujících motorů z palety.
- **Stanice odvozu**- motory jsou odebrány z palet a palety sundány z dopravníku. Nebo je paleta přesunuta na připravený vozík a odvezena.



Obr. 2.1: Dopravníkový systém-náčrtek



Obr. 2.2: Dopravníkový systém-model

2.2 Požadavky zákazníka na řízení dopravníkového systému

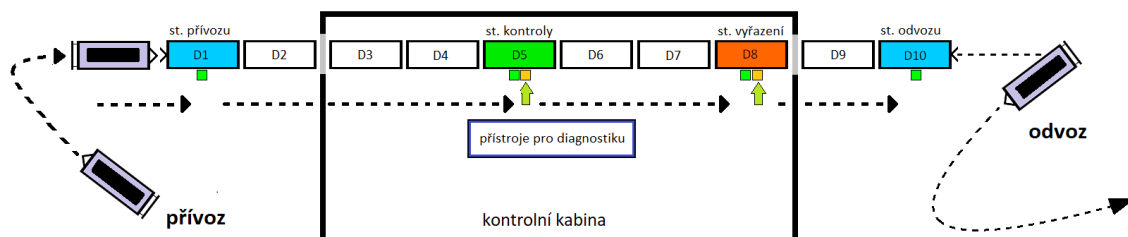
Předně bude definována funkce tak jak ji nadefinoval zákazník. Případné doplňující požadavky vzniknou až po odzkoušení dopravníkového systému v praxi. Je tedy možné, že zákazník něco ohledně funkce ještě doplní. Nebo ji bude chtít trochu pozměnit. Tyto doplňky funkce (modifikace) jsou dále v kapitole 6.3

Celá funkce dopravníkového systému začíná přívozem palety na vozíku. Z vozíku je poté paleta přesunuta na začátek linky. Existuje alternativa a tou je položení prázdné palety na začátek linky a položení prvků na ní až když je na lince. Případně se nakládá paleta již s prvky, ale to je velice nepravděpodobný stav.

Poté se paleta dopraví na druhou stanici konkrétně stanici kontroly. Zde se zastaví a pojedí dál až po odeslání. Odeslání bude realizováno pomocí tlačítka na ovladači. Na tomto místě se vykonává kontrola motorů (diagnostika). Po kontrole obsluha tlačítkem odešle paletu dál po lince.

Po odeslání ze stanice kontroly paleta putuje dále po lince a zastaví se na stanici vyřazení. Zde obsluha poškozené resp. vlastnostmi nevyhovující motory vyřadí. Sundá z palety na dopravníku mimo. Po dokončení, odešle pomocí tlačítka paletu dále po lince.

Paleta se transportuje na konec linky, tedy poslední stanici odvozu. Zde se na posled zastaví a stejně jako na začátku linky je několik možností jak bude s paletou naloženo. První možnost je přesun palety s motory na vozík a poté převoz na další místo. Další možností je sundání motorů z palety a poté sundání samotné palety pryč z linky.



Obr. 2.3: Dopravníkový systém-náčrt funkce

Základy funkce je třeba ještě doplnit. První z vlastností linky bude hromadění palet za sebe automaticky. Tedy pokud je paleta na dopravníku číslo 3 a dopravník číslo 4 je volný pak se paleta automaticky posune na další dopravník. Tak to dělají všechny dokud nenarazí na něco co je zastaví. Tím je myšleno že narazí na zaplněný dopravník před ním nebo na dopravník kde je pro přejezd na další dopravník potřeba

nějaké povolení. Povolení ve formě stisknutí tlačítka na stanicích kontroly a vyřazení.

Bezpečnost

U této linky není třeba řešit bezpečnost pro osoby linku obsluhující po celé délce linky. Vše je v tomto ohledu řešeno pomocí zakrytování nebezpečných částí. Nebezpečné situace nastávají na začátku a konci linky. Zde dochází k přesunu nebo sundávání palet. Musí být zajištěno, že nedojde k pádu resp. sesunutí palety z dopravníku.

Ještě je důležité zajistit, aby nedošlo při diagnostice k přesunu palety dál po lince. Není přípustné aby se paleta s motorem pohnula jakmile s k němu připojí diagnostické přístroje. Je nutné zabránit poškození diagnostickému zařízení. Aby se předešlo delšímu časovému pozastavení provozu linky.

Požadované ošetření poruchových stavů

Pro dopravníkový systém je nutné ošetřit poruchové stavy. Samozřejmě programátor musí počítat se všemi možnými stavy do kterých se jeho systém může dostat. A také počítat s nevyzpytatelným lidským faktorem. Poruchy lze tedy rozdělit na dvě skupiny a to poruchy způsobené nedokonalostí programu a ty způsobené vnějším vlivem ať už člověkem nebo například zaseknutím převážené palety o spoj dvou dopravníků. Zde bude pouze rozebráno co zákazník nadefinoval jako poruchové stavy. Tedy stavy které by neměly či nikdy nesmí nastat. A taktéž nadefinoval, s malou radou programátorů, co se má v případě těchto stavů stát případně vykonat.

Při poruchovém stavu bude vždy celý dopravníkový systém zastaven, aby se zabránilo škodám nebo nedošel pracovník v kontrolní budově úhony. Dále musí být zajištěno, že po odstranění chyby a spuštění linky, nedojde k nesmyslnému posunu palet mimo dopravníkový systém, nebo přesunu palety z kontrolního stanoviště na další stanoviště. Při tomto stavu musí znít výstražná siréna a kontrolní maják musí blikat červenou barvou, aby bylo jasné, že je něco v nepořádku. Stav poruchy bude možné resetovat. Resetovací tlačítko musí být umístěno tak, aby pro jeho stisknutí musel pracovník odstoupit od dopravníkového systému.

Požadavky na řízení

Z pohledu řídicího zařízení. Bylo hlavním požadavkem zákazníka spolehlivost a nízké náklady. Proto bylo k řízení vybráno PLC, konkrétně Siemens CPU 1214C DC/DC/DC.

Z druhého pohledu požadavků na řízení a tedy ovládání lidmi (obsluhou). Byly požadavky definovány v podkapitole číslo 2.2.

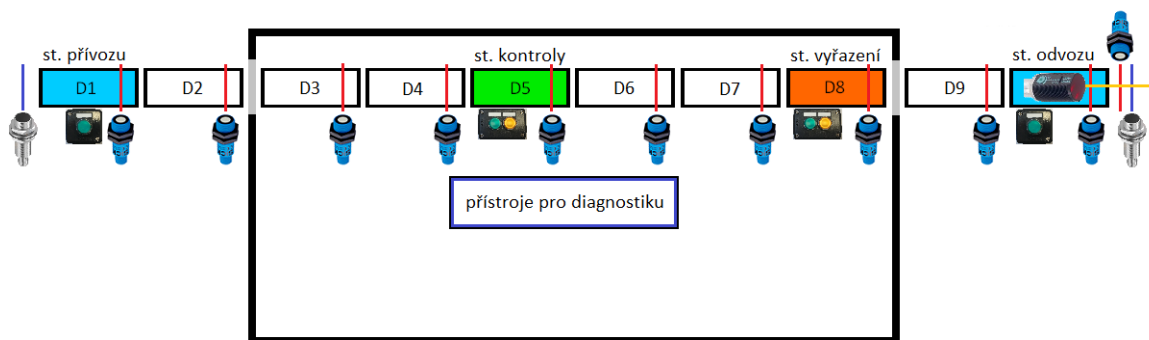
2.3 Umístění čidel, akčních členů a ovládacích prvků

Konkrétní použitá čidla, akční členy a ovládací prvky jsou detailněji popsány v kapitole číslo 5 na straně 21.

Čidla

K dispozici máme čidla, která hlídají přítomnost palety na jednotlivých dopravnících. Dále čidla pro přítomnost vozíku pro přívoz či odvoz a čidlo jištění palety na vozíku. Umístění těchto čidel je znázorněno na obrázku 2.4, kde modrá čidla jsou ta, která snímají přítomnost palety na dopravníku a na D10 je čidlo jištění palety.

Čidla pro zaaretování vozíku na obou koncích linky a čidla pro polohu blokace konce a začátku linky. Na konci linky je oproti začátku navíc čidlo snímající stav blokace palety na vozíku a čidlo přejezdu. Tato čidla na obrázku č. 2.4 nejsou znázorněna.



Obr. 2.4: Dopravníkový systém-umístění čidel a ovladačů

Akční členy

Akčních členů jsou dva typy. Jedním jsou motory a druhým je pneumatický píst. Motory ovládají chod dopravníků, každý dopravník má svůj vlastní motor. Pneumatické písty jsou umístěny na začátku a konci linky. Ovládající blokaci začátku a konce linky, která má za úkol bránit nežádoucímu sjezdu palety z dopravníku. Druhou funkcí je aretace vozíku.

Ovládací prvky

Hlavním ovladačem je rozvaděč, kde se celá linka zapíná a resetuje se případná porucha. Pro ovládání funkce linky jsou rozmístěny ovladače. Na místě přívozu a odvozu jedno tlačítko. A na kontrolní a vyřazovací stanici jsou dvě tlačítka. Dalším ovládacím prvkem jsou tlačítka nouzového vypnutí. Tato tlačítka jsou u všech stanovišť hned vedle ovladačů a na rozvaděči.

3 Programovatelný automat

3.1 PLC Siemens

PLC existuje několik značek, my se zaměříme na značku Siemens. Tato značka svoji řídicí technologii pojmenovala SIMATIC. Vytvořili několik řad, kde každá má své specifické určení. Základní řídicí systém SIMATIC S7-1200 pro automatizační úlohy malého až středně velkého rozsahu, pokročilý systém SIMATIC S7-1500 pro úlohy středně velkého rozsahu až po komplexní aplikace, distribuovaný automat SIMATIC ET 200 pro distribuované systémy a softwarový PLC SIMATIC S7-1500 pro úlohy na platformě PC. [15]

Pro *Programable logic controller* (PLC) Siemens se nedá říct, že by v nějakém směru výrazně vynikalo oproti jiným značkám. Každá značka má své lepší i horší stránky. Obecně lze snad pouze říci, že Siemens značka je nejrozšířenější.



Obr. 3.1: Fotka PLC v rozvaděči

3.1.1 Siemens CPU 1214C DC/DC/DC

Hlavní částí je řídicí reléový modul z řady SIMATIC S7-1200, existují tři třídy výkonnosti (CPU 1211C, CPU 1212C a CPU 1214C). Lze je rozšířit o analogové a digitální moduly pro vstupy/výstupy. Důležité parametry jsou uvedeny v tabulce 3.4. [5]

Cena se pohybuje 21500 Kč za jeden kus.

3.1.2 Siemens SM 1221

Jedná se o rozšiřovací modul z řady SIMATIC S7-1200, kterého byly použity dva kusy. Tento modul je určený pro digitální vstupy (*digital input* (DI)). Nabízí nám 16 digitálních vstupů na 24 V/DC. Prodává se s různým počtem vstupů, verze 8 a 16 vstupů. Protože pracuje se vstupními signály budou k němu tedy připojena všechna použítá čidla. Důležité parametry jsou uvedeny v tabulce 3.5.

Cena se pohybuje 7990 Kč za jeden kus. [7]



Obr. 3.2: SM 1221 [7]

3.1.3 Siemens SM 1222

Jedná se o rozšiřovací modul, kterého jsou použity 2 kusy. Tento modul je určený pro digitální výstupy (*digital output* (DO)). Nabízí nám 16 digitálních výstupů na 24 V/DC. Tento typ se prodává ve dvou variantách, kde je buď 8 nebo 16 výstupních svorek. Protože pracuje s výstupními signály, tedy posílá signály hodnot true/false(24 V/0 V) ven, budou k němu připojeny například: motory, blokace, kontroly. Důležité parametry jsou uvedeny v následující tabulce 3.6.

Cena se pohybuje 6390 Kč za jeden kus. [8]



Obr. 3.3: SM 1222 [8]

Rozhraní	RJ45 integrovaný, RS 232, RS485 volitelně
Počítadlo	3 (100 kHz) + 3 (30 kHz)
Počet výstupů relé max.	10
Počet analogových vstupů	2
Maximální počet vstupů	16
Počet digitálních vstupů max.	14
Počet výstupů max.	1
Podporované protokoly	TCP/IP, ISO-on-TCP, USS drive protocol, \\ Modbus MAster/Slave
Provozní napětí (max.)	28.8 V/DC
Provozní napětí (min.)	20.4 V/DC
Provozní napětí (rozsah)	20.4 - 28.8 V/DC
Hodiny (provedení)	Přepínání léto/zima
Provozní napětí	24 V/DC
Vnější šířka	110 mm
Vnější výška	100 mm
Programová paměť	2 MB
Krytí	IP20
Datová paměť	50 kB
Počet rošiřující modulů	Max. 1 Signalboard, 8 Signalmodule, \\ 3 Kommunikationsmonudel
Typ (výrobce)	CPU 1214 DC/DC/DC
Číslo výrobce	6ES7214-1AG31-0XB0
Displej	Ne
Kategorie produktu	PLC řídicí modul

Obr. 3.4: Tabulka CPU 1214C DC/DC/DC

Kompatibilní	Siemens SIMATIC S7-1200
Typ	SM 1221
Číslo výrobce	6ES7221-1BH32-0XB0
Provedení	16x digitální vstup 24 V/DC, \\ 16 potenciálově odděleno
Maximální počet vstupů	16
Provozní napětí (rozsah)	20.4 - 28.8 V
Provozní napětí (min.)	20.4 V
Provozní napětí (max.)	28.8 V
Provozní napětí	28.8 V
Vnější výška	100 mm
Vnější šířka	45 mm
Vnější délka	75 mm
Krytí	IP20
Provozní teplota (min.)	-20 °C
Provozní teplota (max.)	+60 °C
Kategorie produktu	Digitální vstup pro PLC

Obr. 3.5: Tabulka Siemens SM 1221

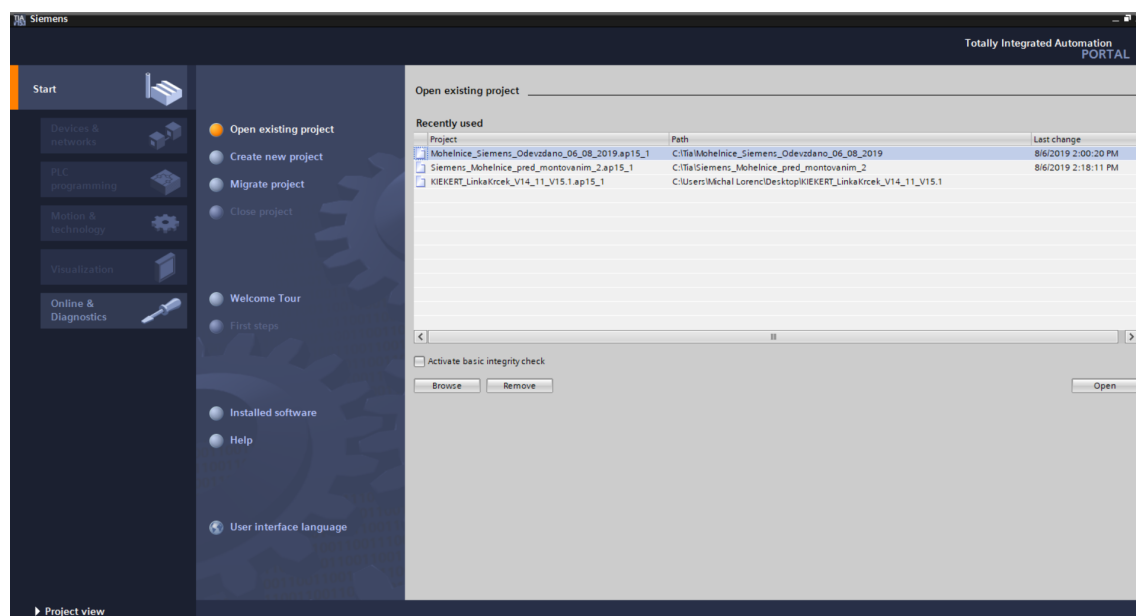
Kompatibilní	Siemens SIMATIC S7-1200
Typ	SM 1222
Provedení	16x digitální výstup 24 V/DC \\ tranzistor 0.5 A
Číslo výrobce	6ES7222-1BH32-0XB0
Počet výstupů max.	16
Provozní napětí (rozsah)	20.4 - 28.8 V
Provozní napětí (min.)	20.4 V
Provozní napětí (max.)	28.8 V
Provozní napětí	28.8 V
Vnější délka	75 mm
Vnější výška	100 mm
Vnější šířka	45 mm
Krytí	IP20
Provozní teplota (min.)	-20 °C
Provozní teplota (max.)	+60 °C
Kategorie	digitální výstup pro PLC

Obr. 3.6: Tabulka Siemens SM 1222

4 Vývojové prostředí Tia Portal

Co je to Tia portal? Jedná se o software od společnosti Siemens, základně určený pro programování PLC. Rozepsáním zkratky dostaneme **zkTia!** (**zkTia!**). Poskytuje vše pro potřeby automatizace. V **zkTia!** je možné konfigurovat, diagnostikovat a v neposlední řadě programovat řídicí systémy, elektrické pohony, ovládací prvky, systémy zajišťující funkční bezpečnost stroje atd. Hlavní výhodou, kterou tento software přinesl je skutečnost, že všechna data jsou uložena a zpracována na jednom místě. Proto se s nimi pracuje mnohem snáze, již není nutné zpracovávat data v několika odlišných softwarech a poté je spojovat.[9]

Na obrázku č. 4.1 vidíme vstupní stránku portálu, kde máme na výběr z projektů na kterých pracujeme.



Obr. 4.1: Prostředí Tia Portal

4.1 Programovací jazyky

Programovací jazyky se liší v závislosti na výrobci PLC a prostředí v kterém ho programujeme. Mezi jazyky není nijak velký rozdíl. Liší se spíše v detailech, s trochou nadsázky lze říci, že přechod mezi jazyky není velký problém. Konkrétně značka Siemens a prostředí Tia portal využívá následující jazyky.

STL - statement list

Statement list neboli seznam instrukcí. Jde o textový jazyk, kde se kód skládá ze sekvence instrukcí, každá instrukce je vložena na nový řádek.

SCL - structured control language

Structured Control Language jedná se o textový jazyk, který se hodí k složitějším výpočtům a algoritmům. Kde by grafický jazyk byl velice nepřehledný. Celý kód je psán tzv. strukturovaným textem.

LAD - ladder diagram

Ladder Diagram grafický typ jazyka. Programování probíhá pomocí reléových schémat. Vzhledem může připomínat kreslení schémat zapojení. Za výhodu lze považovat v případech jednodušších algoritmů přehlednost a názornost řešení algoritmu. Za nevýhodu naopak nepřehlednost při větším rozsahu kódu.

FBD - function block diagram

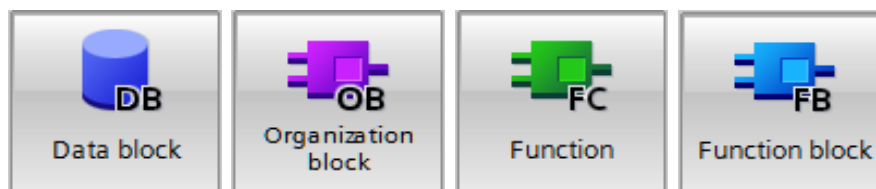
Function Block Diagram, programování pomocí funkčních bloků, opět se jedná o grafický jazyk. Programování probíhá spojováním funkčních bloků za sebe, logika skládání bloků je stejná jako u LAD kde se jde zleva doprava. Opět díky tomu, že se jedná o grafický jazyk je velice názorný při lehčím kódu.

GRAPH

Je posledním grafickým jazykem. Název přesně odpovídá jeho vzhledu. Mohl by být charakterizován jako znázornění stavového automatu, kde se přepínáme stav ze stavu za nějaké podmínky. Nevýhodou je zpracování složitého algoritmu, kde jsou potřeba postranní výpočty.

4.2 Programovací bloky

Programování se zde dá rozložit do základních bloků. Hlavní částí je blok, který se vytvoří automaticky při založení projektu OB1. Dále budou specifikovány ostatní bloky.



Obr. 4.2: Ikony jednotlivých bloků

OB - organizační blok

Organizačních bloků jsou dva základní typy automaticky vytvořené při založení projektu. Prvním je OB1, což je hlavní blok, který se cyklicky opakuje. Nelze ho odebrat je to stavební kámen celého programu. Druhým blokem je Startup neboli OB100, který neběhá cyklicky, je spouštěn pouze při prvním spuštění nebo restartu *central processing unit* (CPU).

DB - data blok

Datovým blokem rozumíme místo pro uložení proměnných. Náhradou tvorby DB je tvorba tagu, který je poté v tag tabulce. Proměnné v obou případech mohou být různých typů.

FB - funkční blok

Jedná se o blok s (vlastním) kódem a může ale nemusí mít vstupy a výstupy. Jedná se vlastně o funkci s tím rozdílem, že FB si vždy při založení vytvoří vlastní DB což nám umožňuje mít nějaké vnitřní proměnné a nepracovat pouze s vnějšími. Tento blok má vždy alespoň jeden vstup a tím je vstup enable pro jeho zapnutí.

FC - funkce

Obdoba FB s rozdílem, že pracuje pouze s vnějšími proměnnými. Má nějaké vstupy a výstupy, kterým se přiřadí dané proměnné.

4.3 WinCC

WinCC je nástroj pro tvorbu vizualizace v tia portal. Vizualizace umožňuje nasimulovat si řízenou technologii a její chování. To lze poté promítat na HMI panel, kde je možné řízenou technologii případně ovládat nebo pouze sledovat aktuální dění. Vizualizace lze také udělat pouze pro prohlížení v počítači v Tia portal.

5 Komponenty řízené technologie

5.1 Čidla

Sick UM18 / UM18-2 Core

Jedná se o ultrazvukový snímač, který pracuje na principu time of flight. Tedy vyšle ultrazvukový signál a čeká na jeho odražení. Toto odražení je přijato a z časového rozdílu mezi vysláním a přijmutím signálu je výpočtem stanovena vzdálenost objektu. V případě tohoto konkrétního typu, ale neměříme vzdálenost pouze přítomnost překážky v rozsahu čidla. [13]

Ultrazvukový snímač Sick UM18 byl vybrán kvůli jeho ceně, měřicímu rozsahu a šířce měřicího laloku. Šířka laloku je totiž určena vzdáleností dvou válečků mezi kterými je čidlo umístěno. Válečky nesmí spadat do měřicího rozsahu.

Balluf BES M18MI-PSC12B-S04

Indukční snímač pracující s vnitřní cívkou, která si tvoří vlastní magnetické pole. Pokud je toto pole narušeno elektricky vodivým materiálem dojde ke změně indukčnosti cívky. Toto čidlo vrací pouze signál logické úrovně 1 nebo 0. Bylo vybráno pro svou jednoduchost, nízkou cenu a spolehlivost co se životnosti týče. [14]MI-PS

Indukční snímač Balluf BES M18MI-PSC12B-S04 byl vybrán pro jeho cenu, měřicí rozsah. Snímač je umístěn tak, že okolo sebe má hodně věcí, které by mohl snímat, proto je zvolen s malým měřicím rozsahem.

Festo SMT/SME-8

Jedná se o polovodičový automatický spínač z řady D-M9N, typem PNP. Použitý typ má konektor M8-3pin a 0,5 m kabel. Obsahuje řadu statických hlásičů s integrovanými přípojkami. Díky tvaru zajišťuje snadné nasazení do drážky válce. Má snížený zatěžovací proud. Hodí se pro relé, tištěné obvody, programovatelné řídicí jednotky.[16]

Čidlo Festo SMT/SME-8 bylo vybráno pro jeho nízkou cenu.



Obr. 5.1: Čidlo Sick [13]



Obr. 5.2: Čidlo Balluf [14]



Obr. 5.3: Čidlo festo [16]

Pepperel GLV18-8-H-120/73/120

Triangulační optické čidlo s potlačeným pozadím. Rozsah snímání je 10-120 mm, vyzařovací úhel je 2°. [17]

Čidlo Pepperel GLV18-8-H-120/73/120 bylo vybráno protože má potlačené pozadí. Vzhledem k potřebnému měřicímu rozsahu není možné použít indukční čidlo.

5.2 Akční členy

Siemens 1LE1002-0CB22

Jde o typ asynchronního elektromotoru s výkonem 0.25 kW. V požadavcích zákazníka bylo, že výrobcem motoru musí být Siemens. Tento konkrétní typ byl zvolen pro tento dopravník, protože jeho výkon a otáčky jsou dostačující. Posledním parametrem byl také poměr cena výkon. Výstupní otáčky z motoru jsou zpřevodovány na malé otáčky, aby dopravníky šly pomalu.

Tato konkrétní značka byla požadována zákazníkem.

Festo ADN-32-80-A-P-A

Jedná se o dvojčinný píst tlačенý vzduchem. Zdvih je 80 mm to je pro použití dostačující. Pro aretaci vozíku i pro blokaci na konci linky. [18]

Festo ADN-32-80-A-P-A byl zvolen protože je poháněn vzduchem a dodavatel s jeho použitím má zkušenosti.



Obr. 5.4: Elektromotor [11] Obr. 5.5: Pneumatický píst [18] Obr. 5.6: Čidlo Pepperel [17]

5.3 Ovládací prvky

Rozvaděč

Obsahuje několik kontrolky a tlačítek (některá podsvícená). Konkrétně kontrolky ODARETUJ EM. STOP, 24V DC, ZAPNUTO, CHOD AUT, PORUCHA. Tlačítka ZAPNI, START, VYPNI, STOP a KVITACE PORUCHY. A nouzové tlačítko NOUZOVÉ ZASTAVENÍ. Samozřejmě je na rozvaděči ještě hlavní vypínač.

Nás nebudou v softwaru zajímat kontrolky ODARETUJ EM. STOP, 24V DC, ZAPNUTO a to protože nejsou ovládány softwarem. A tlačítka ZAPNI a VYPNI, která rovněž nejsou napojena na vstupy PLC.



Obr. 5.7: Fotografie části rozvaděče

Ovladače

Jsou použity dva typy oba po dvou kusech. První pár má dvě podsvícená tlačítka a druhý pár má jedno podsvícené tlačítko. Ovladač s dvěma tlačítky, kde jedno je žluté a druhé zelené, má popisky ODESLAT a BLOKACE. Ovladače s jedním tlačítkem mají popisky ODESLAT a PŘIVOLAT. Přivolat je na začátku linky a odeslat na konci.

Tento typ ovladače byl vybrán pro jednoduchost a robustnost.



Obr. 5.8: Fotografie ovladače na stanici kontroly

6 Software

6.1 Návrh koncepce řídicího programu

Software bude rozvržen na několik částí, z nichž každá bude popsána co bude obsahovat. A budou nastíněny hlavní problémy, které budou řešeny. Tedy přívoz, odvoz a přesuny palety po lince. Dále ovládání, signalizace a bezpečnost.

Rozdělení programu do dílčích bloků

Dle rady zkušenějších bude program dělen na několik základních bloků. Prvním blokem bude **stav linky**. V tomto bloku bude kontrola všech prvků, které zajišťují bezpečnost nebo brání správné funkci v případě poruchy. Těmito prvky rozumíme: nouzová tlačítka, hlavní vypínač, PLC napájení, jističe, pojistky, přepětová ochrana. Pokud budou všechny v pořádku bude třeba to nějakou proměnou reprezentovat dále v programu. Aby v případě že něco selže jsme tuto proměnnou změnili a dále v programu věděli, že je něco v nepořádku a musíme zastavit funkci. Musí tedy z tohoto bloku vystupovat informace o "stavu linky" (je-li vše v pořádku).

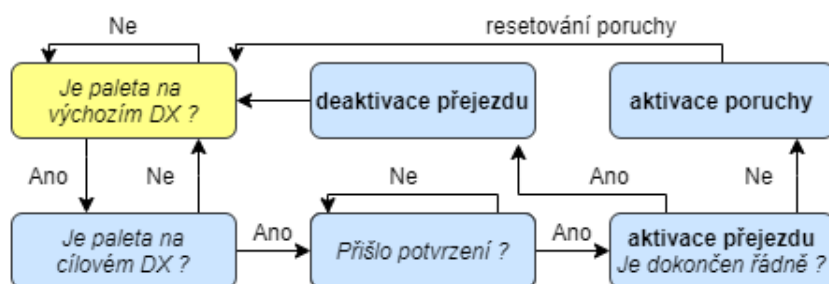
Dalším blokem bude **signalizace** zaměřená na ovládání všech signalizačních prvků. Už je stvořena kontrola prvků a v případě, že je vše v pořádku resp. není, je třeba to signalizovat. K tomuto účelu využijeme kontrolek, prosvícených tlačítek a signalizačního sloupku včetně sirény. Zde se budou řešit podmínky za jakých se má určitá konkrétní signalizace zapnout resp. vypnout. Například pokud bude jeden jistič shozený, první blok zahlásí chybu a chyba v tomto bloku sepne signalizaci pomocí kontrolky označenou porucha na rozvaděči.

Bude vytvořen blok **automatický chod**, který bude řídit celý dopravníkový systém. Bude řešit přívoz a přesun palety na dopravník, přesun palety z dopravníku na vozík a odvoz. A přesouvání palety přes celou linku, přes všechna stanoviště za určitých podmínek. U všeho musí řešit bezpečný provoz, především u přesunů palet z a na vozík, aby pracovník nebo prvek z palety nedošel jakékoli úhony.

Vývojový diagram hlavního bloku

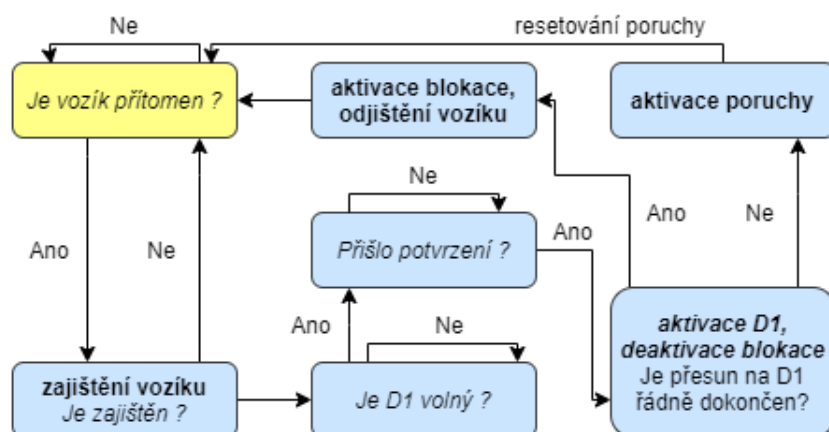
U všech diagramů je cesta jak se dostat do stavu poruchy. Jedná se o jištění případu, kdy se něco neodehraje řádně. Následkem čehož je třeba zabránit dalším chybám. Tento stav bude pracovat s vnější proměnou, která bude signalizovat poruchový stav. U všech diagramů je výchozí stav vyznačen žlutou barvou. Bude počítáno že z tohoto stavu se začíná a vrací se do něj, ať se stane chyba nebo se akce provede bez komplikací.

Na prvním diagramu je znázorněn princip přejezdu palety mezi jednotlivými dopravníky. Protože dopravníků je 10 a přejezd se bude konat ten samý několikrát, bude vhodné realizovat ho pomocí funkčního bloku (FB). Aby byl diagram aplikovatelný na všechny přejezdy, včetně těch kde je třeba aby k přejezdu nedošlo automaticky ale po stisku tlačítka. Je v diagramu zařazeno potvrzení. To je nutné realizovat tak aby bylo možné potvrzení vyřadit v případě že se má přejezd konat automaticky.



Obr. 6.1: Diagram pro přejezd mezi dopravníky

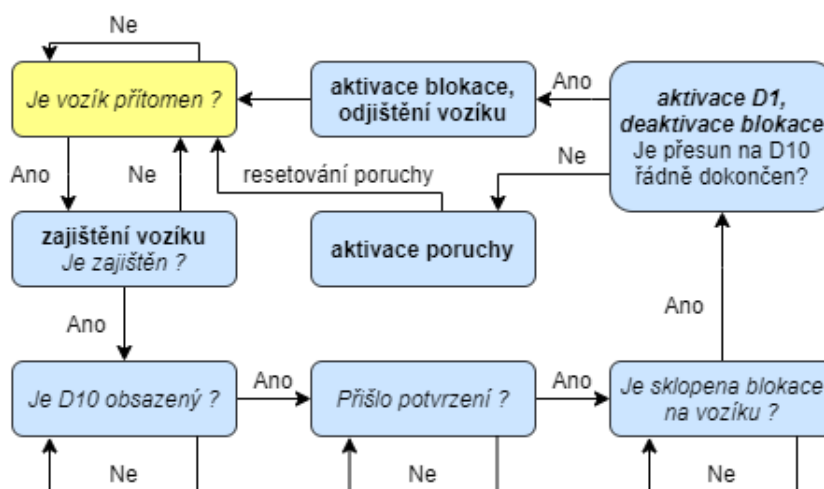
Diagramy 6.2 a 6.3 jsou zaměřeny na problematiku přesunu palety na resp. z dopravníku. Prvním diagram je pro přívoz palety na vozíku, kde dochází k jejímu přesunu na první dopravník. Tento přejezd je jednodušší o čidlo jištění palety na vozíku, které na začátku linky není.



Obr. 6.2: Diagram pro přívoz palety na D1

Na posledním diagramu je znázorněn přesun palety z konce dopravníku na vozík. V diagramu je znázorněno, že před aktivací přejezdu musí být splněny určité podmínky. Ale už v diagramu není řešen problém kdyby nějaké čidlo selhalo nebo například vypadlo zajištění vozíku, když už jsme programově v části diagramu za zajištěním vozíku. Tento problém bude muset být řešen, aby tyto nutné podmínky pro přejezd byly kontrolovány a splňovány nepřetržitě. Pokud by například došlo k selhání jistění vozíku je nezbytné resetovat přejezd a zabránit pádu palety mimo dopravník a vozík.

U všech diagramů platí, že stav vyznačený žlutě je uvažován jako výchozí stav.



Obr. 6.3: Diagram pro odvoz palety z D10

6.2 Implementace

Při programování pomocí jakéhokoli jazyka je nutné mít kód přehledný, aby v případě úprav cizí i vlastní osobou, byl člověk schopný se zorientovat. Proto je nutné kód členit na co možná nejvíce jednoduchých úseků. Protože se jedná o tvorbu pod určitou firmou, podléhá kód firemní konvenci. Je to pro případ nehody nebo úpravy funkčnosti, kterou by nebyl schopný vykonat původní tvůrce kódu. Proto se kód člení na určité bloky, vstupy a výstupy se pojmenovávají dle konvence. Dělíme program na bloky a krátké networky pro přehlednost.

Oživení technologie

Nejprve byla provedena hardwarová konfigurace. Propojení počítače s PLC. Přidání správného typu PLC do projektu. Pojmenování PLC vstupů a výstupů. To bylo provedeno dle firemní konvence, které program musel podléhat. Například čidla začínají c_BQ1. Po konfiguraci byla provedena kontrola všech čidel, kontrollek, tlačítek...

Díky watch table v tia portal je možné monitorovat jakou čidlo vrací hodnotu. Tak postupným zakrýváním a odkrýváním čidel byla všechna čidla zkontrolována. Nefunkční byla opravena či nahrazena a všechna seřízena pro řádnou funkci. Dále byla provedena kontrola funkčnosti všech motorů a pneumatických válců. K tomuto účelu nám v tia portal posloužila force table. Kde byly postupně ovládány jednotlivé motory a pneumatické válce .

Ukázalo se že rozvaděč není správně zapojen. Naštěstí se nejednalo o chybu, která by po zapnutí něco zničila. Tlačítko ZAPNI, VYPNI a STOP na rozvaděči neplnilo svou funkci. Ukázalo se, že je špatně zapojeno ovládání napájení PLC. Dále bylo přepojeno vzduchové ovládání blokace palet proti pádu z konců linky A to tak, že při výpadku napájení bude blokace aktivní a nedojde k pádu palety. Tedy signál hodnoty false bude znamenat aktivní blokaci.

Při manipulaci s vozíkem s naloženou paletou se zjistilo, že vozík je příliš velký na paletu. A paleta na něm se samovolně pohybuje. To bylo řešeno jištěním na konci vozíku. To ale zabráňuje pouze sjetí palety, ale nikoli jejímu nárazu do jištění a poté možnému pádu motorů z palety. Proto bylo nutné vozíky náležitě upravit.

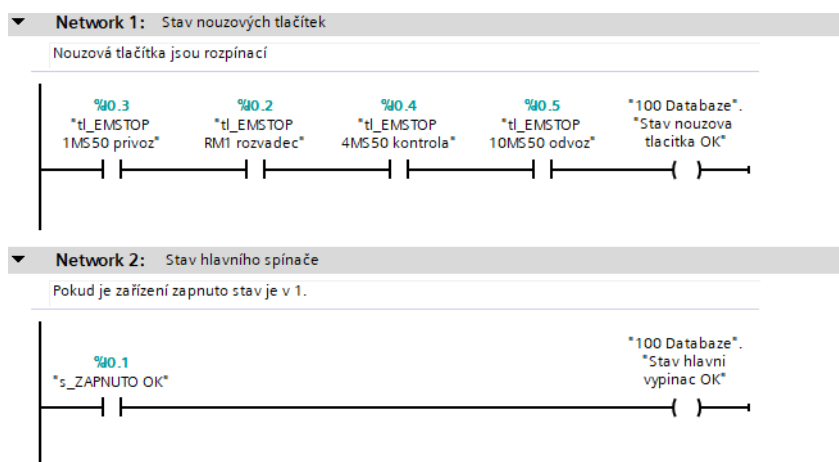
6.2.1 Stav linky

Nejprve začala implementace bloku "stav linky", protože stav čidel, motorů, kontrol lze pozorovat v online režimu. V dokumentaci byly dohledány všechny kontrolní prvky pro kontrolu v bloku stav linky. Do jednotlivých networků byly dány jednotlivé druhy kontrolních prvků. Například viz obrázek 6.4 stav nouzových tlačítek.

Kontroly jednotlivých prvků budou separované, každá bude mít svůj vlastní network. Aby při potřebě kontroly stavu jedné věci stačilo přepnout na konkrétní komponentu. V programu není nutné při více komponentách jako jsou jističe, je dávat paralelně jeden jistič na jeden řádek (rung) a to ani za účelem přehlednosti. Protože v případě kontroly který jistič je shozen, budou aktivní jističe svítit zeleně a jističe neaktivní/shozené budou svítit červeně.

Jakákoli z těchto poruch bude nastavovat bit na 1 (setovat) jeden konkrétní bit s názvem porucha. Tento bit bude informativním bitem pro kontrolu zda nenastala porucha. Bude využíván i dále v programu, protože porucha může nastat mnoha způsoby. Například při přejezdu palety, když paleta za určitý čas nedojede na místo určení je jasné, že je něco v nepořádku. Zaseknutí palety nebo zadrhnutý dopravník a podobně.

Kód bude v tomto případě bude velice jednoduchý. Vezměme například nouzové tlačítka. Zde bude použit spínací kontakt s adresami jednotlivých nouzových tlačítek a na konci řádku bude umístěna funkce SET s proměnou porucha. Toto bude provedeno pro všechny komponenty.



Obr. 6.4: Blok stav linky - program

6.2.2 Signalizace

Pro obsluhu pracoviště nejdůležitější část programu, protože zajišťuje pomyslnou komunikaci mezi strojem a pracovníkem. Všechna signalizace musí být správně uvedena v návodu k obsluze zařízení. Ideálně i s pokyny jak který stav řešit. Pokud se jedná o složitější problém.

Pro signalizaci jsou použity kontakty normaly open, které mají přiřazeny příslušné proměnné. Za ně připojíme assignment která je sepnutá pokud je napájena. Výsledkem toho by bylo svícení kontrolky dokud se stav příslušné proměnné nezmění opět na 0. Pokud je vyžadováno blikání, bude do série připojen ještě clock memory bit. Tento bit s určitou frekvencí mění svůj stav z 0 na 1. Pro jejich vytvoření stačí přidat normaly open kontakt a začít psát clock. Poté se nám nabídne několik signálů s různou frekvencí, která je zapsána v názvu.

Kontrolkám na rozvaděči byly přiřazeny následující funkce:

- **Odaretuj EM.STOP** - svítí v případě, že je zamáčknuté alespoň jedno tlačítko nouzového vypnutí
- **24 V DC** - svítí pokud je zapnuto napájení PLC (24 V)
- **Zapnuto** - svítí při zapnutém napájení ovládání
- **Chod aut.** - svítí pokud je dopravníkový systém v provozu (automatickém chodu), bliká pokud se linka připravuje na provoz (trvá po čas inicializace)
- **Porucha** - bliká v případě poruchy

Kontrolkám na jednotlivých stanovištích byly přiřazeny následující funkce:

- **stanice kontroly**
 - odešli - svítí pokud je možné paletu odeslat, bliká pokud se paleta odesílá na další dopravník
 - blokuj - nevyužito
- **stanice vyřazení**
 - odešli - svítí pokud je paletu možné odeslat, bliká pokud se paleta odesílá
 - blokuj - nevyužito
- **začátek dopravníku (přívoz palet)**
 - odešli - svítí pokud je možné přesunout paletu na dopravník, bliká pokud přesun probíhá
- **konec dopravníku (odvoz palet)**
 - odešli - svítí pokud je možné paletu přesunout na vozík, bliká pokud přesun probíhá



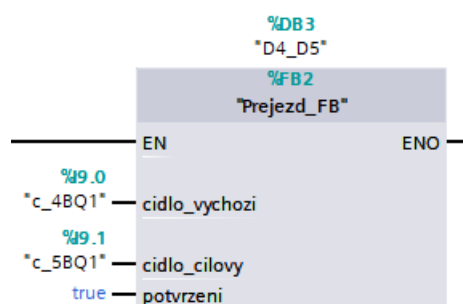
Obr. 6.5:
Signalizační
sloupek
[12]

6.2.3 Přejezd

Protože na celé lince jsou přejezdy mezi částmi dopravníků principiálně shodné byl vytvořen funkční blok pro přejezd mezi jednotlivými dopravníky a funkční blok pro přejezd z vozíku na dopravník nebo z dopravníku na vozík. Předně je nutné definovat, co bude pro jednotlivé přejezdy jiné a co budou vstupy a výstupy funkčních bloků. Oba bloky budou rozebrány odděleně.

Přejezd mezi jednotlivými dopravníky-A

Vstupními parametry budou čidla přítomnosti palety na jednotlivých dopravnících. Tím bude řešena kontrola možnosti vykonání přejezdu. Je paleta výchozí plná a cílová volná lze přejezd provést. U některých přejezdů je potřeba ještě potvrzení (například přejezd ze stanice kontroly). Toto potvrzení je provedeno stlačením tlačítka obsluhou. Proto vstupním parametrem bude ještě potvrzení. Pokud by tento vstup nebyl potřebný stačí při volání bloku nastavit na vstup "true" jako pevnou hodnotu.



Obr. 6.6: FB přejezd A

Ovládání motorů jednotlivých dopravníků je možné řešit výstupní proměnou nebo vnitřní proměnou static. Zde bylo použito řešení pomocí vnitřní proměnné typu static. Výhodou je, že blok vypadá jednodušeji a ovládání motorů je možné řídit z jednoho místa. Tedy pokud bychom použili výstupní proměnnou byl by na každý blok připojen i konkrétní pár motorů. Z toho plyne, že jeden motor by byl současně ovládán ze 3 míst. Například motor číslo 3 by byl ovládán přejezdem D2 až D3 a přejezdem D3 až D4 a inicializací. V případě použití static, ale máme ovládání jednotlivých motorů v jednotlivých a separovaných networkích kde je pouze zavolaná proměnná static příslušného bloku přejezdu.

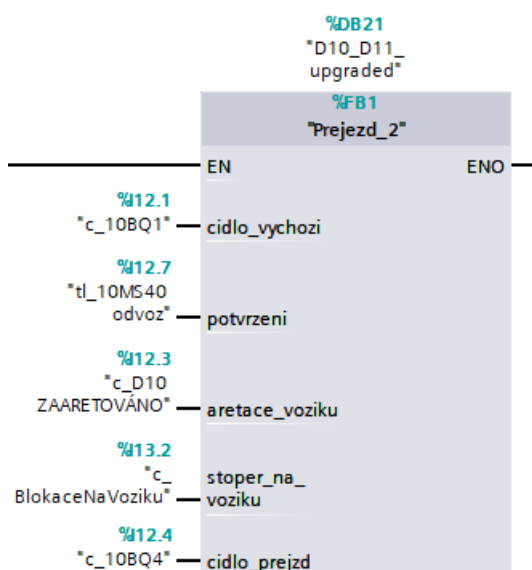
Vnitřní zapojení bloku přejezdu je navrženo dle diagramu 6.1.

Přejezd z vozíku na dopravník

Tento přejezd se sice liší od přejezdu mezi dopravníky, ale není nutné tvořit nový funkční blok. Řešením bylo na vstupní proměnou *potvrzení* přivést sériovou kombinaci čidla aretace vozíku, čidla přítomnosti vozíku a stisknutého tlačítka. Tedy pokud je vozík přítomen, zaaretován a pracovník stiskne tlačítko pro potvrzení přejezdu dojde k spuštění přejezdu. Prosvícené tlačítko pro potvrzení svítí pokud je první dopravník volný a podmínky pro přejetí palety z vozíku jsou v pořádku. Bliká při vykonávání přejezdu na dopravník.

Přejezd z dopravníku na vozík-B

Tento přejezd se liší od převozu palety na dopravník tak, že má navíc čidlo aretace na vozíku a čidlo přejezdu. Je nutné při této operaci ošetřit, aby nedošlo k pádu palety z dopravníku. Proto byl rozšířen blok přejezdu o vstupní proměnné na přítomnost vozíku, aretaci vozíku, čidlo přejezdu (čidlo které snímá zda je paleta už alespoň z části mimo konec dopravníku). Ideální by bylo mít čidlo pro kontrolu přejetí palety na vozíku v pozici kde je jasné že je vše v pořádku. Bohužel to nebylo proveditelné, proto je zastoupeni čidlem přejetí na vozík.



Obr. 6.7: FB přejezd B



Pátý řádek spíná poruchový bit když do stanoveného času neresetuje bit *Spina-*

cPorucha.

Poslední řádek resetuje *SpinacPorucha* a *porucha* po zmáčknutí tlačítka KVI-TACE RESET na rozvaděči.

Na obrázku 6.9 je výčet všech proměnných v bloku přejezdu.

Přejezd_FB								
	Name	Data type	Default value	Retain	Accessible f...	Writa...	Visible in ...	Setpoint
1	▼ Input				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2	cidlo_vychozi	Bool	false	Non-ret...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
3	cidlo_cilovy	Bool	false	Non-retain	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
4	potvrzeni	Bool	false	Non-retain	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
5	▼ Output				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
6	<Add new>				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
7	▼ InOut				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
8	<Add new>				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
9	▼ Static				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
10	prejezd	Bool	false	Non-retain	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
11	timeout	IEC_TIMER		Non-retain	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
12	porucha	Bool	false	Non-retain	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
13	CasFiltrVychodi	IEC_TIMER		Non-retain	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
14	CasFiltrCil	IEC_TIMER		Non-retain	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
15	FiltrPotvrzen	Bool	false	Non-retain	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
16	SpinacPoruchy	Bool	false	Non-retain	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
17	▼ Temp				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
18	<Add new>				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
19	▼ Constant				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
20	<Add new>				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Obr. 6.9: Funkční blok přejezd B - vnitřní proměnné

6.2.4 Automatický chod

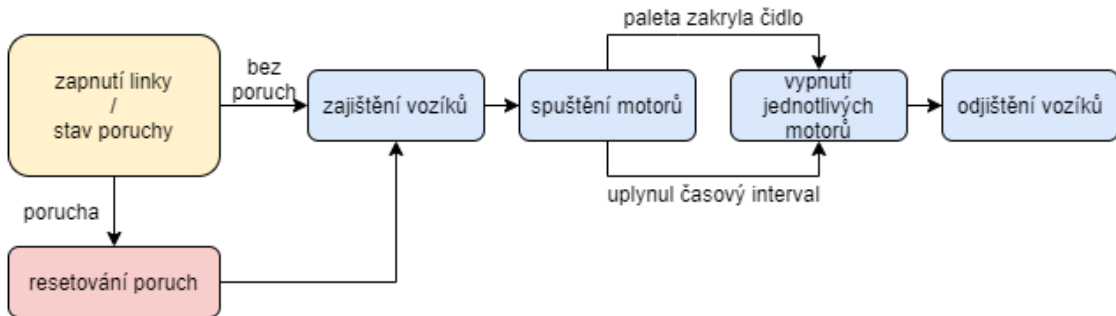
Blok propojuje jednotlivé komponenty, zajišťuje kompletní funkci celého programu. Například obsahuje ovládání motorů, všechny přejezdy na lince, ovládání blokace konců linky, zajišťování vozíku a další.

Inicializace

Při spouštění linky je nutné zjistit zda na ní jsou nějaké palety a případně je dopravit na místo kde je čidla zachytí. Ať už se bude jednat o spuštění linky na začátku směny nebo po nějaké poruše je nutné provést inicializaci, aby se program nezacyklil v nějakém nesmyslném stavu. Inicializace probíhá vždy při přechodu do automatického režimu linky. Po zmáčknutí automatický chod se provede inicializace a poté je zařízení v automatickém chodu.

Nutnou podmínkou pro start inicializace, je odstranění všech poruch. Pokud by byli připojeny vozíky je nutné, aby při inicializaci zůstaly připojené pro případ, že by se vykonával přesun z resp. na vozík. Po inicializaci bude jistění vozíku deaktivováno. Je uvažován stav, že na lince je neznámý počet palet neznámo kde. Dojde ke spuštění všech dopravníků dokud nebudou zaplněny paletou (dojde k zachycení

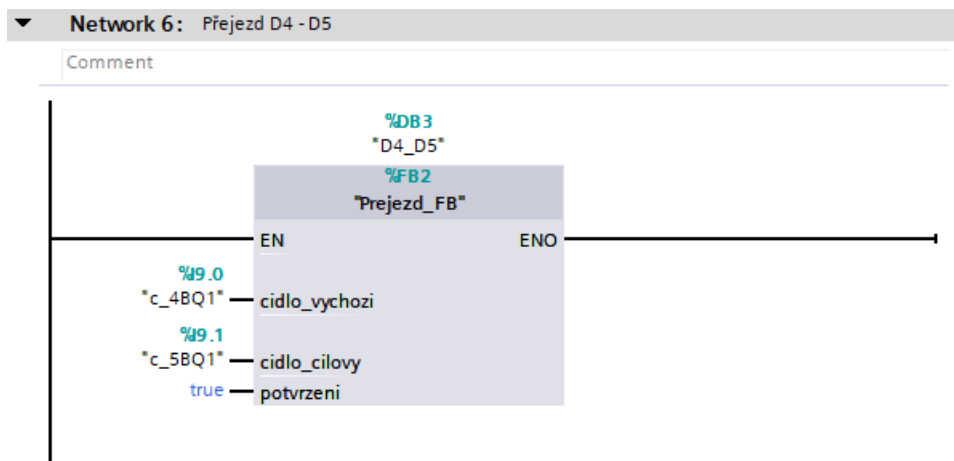
palety čidlem) nebo neuplyne časový interval pro vykonání inicializace. Pokud paleta zastaví na dopravníku, který má před sebou volný dopravník a není určen k zastavení palety do potvrzení odeslání. Bude zde paleta stát a k přesunu dojde až po spuštění automatického chodu.



Obr. 6.10: Diagram průběhu inicializace

Přejezdy

V tomto bloku je volán funkční blok přejezdu na každý přejezd. Jednotlivé instance jsou pojmenovány jako výchozí dopravník_cílový dopravník. D2_D3 je například přejezd mezi D2 a D3. Přejezd na začátek linky je pojmenován D0_D1 a přejezd z konce linky na vozík je D10_D11. Popis přejezdů je na stránce 30.



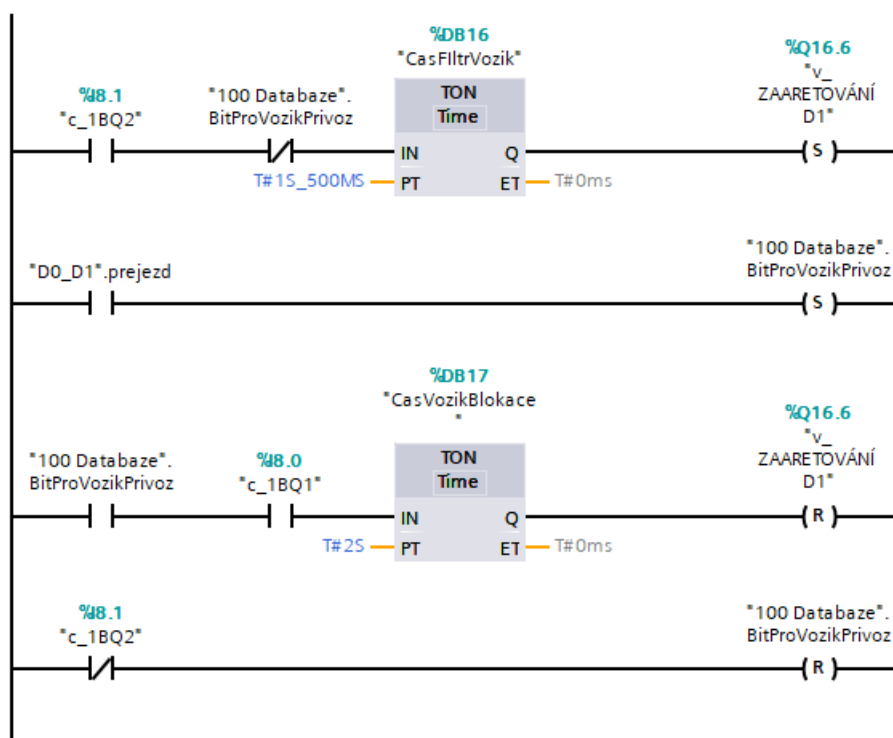
Obr. 6.11: Konkrétní instance bloku pro přejezd mezi D4 a D5

Zajištění a odjištění vozíku na začátku linky - přívoz

Pro přesun palety z resp. na vozík je nutné vozík zajistit. Bylo by možné vnořit funkci zajištění a odjištění vozíku do bloku přejezdu, ale pro jednoduchost bloku a snadnější implementaci je řešení externě.

Na prvním řádku je čidlo *c_1BQ2* přítomnosti vozíku. Pokud je vozík přítomen tak po uplynutí časovače dojde k jeho zajištění. Pomocný bit *BitProVozikPrivoz* je zde, aby rozlišil zda je stav kdy je na lince volno a je tedy žádoucí zaaretovat vozík nebo je linka plná a není možné na ní přesunout další paletu.

Druhý řádek ovládá pomocný bit. Jakmile se aktivuje přejezd, setuje se pomocný bit. Na třetím řádku dojde k odjištění vozíku, pokud se paleta přesune na první dopravník (zachytí ji čidlo), pomocný bit je sepnutý a uplyne doba na časovači. Poslední řádek nám resetuje pomocný bit pomocí čidla přítomnosti vozíku.



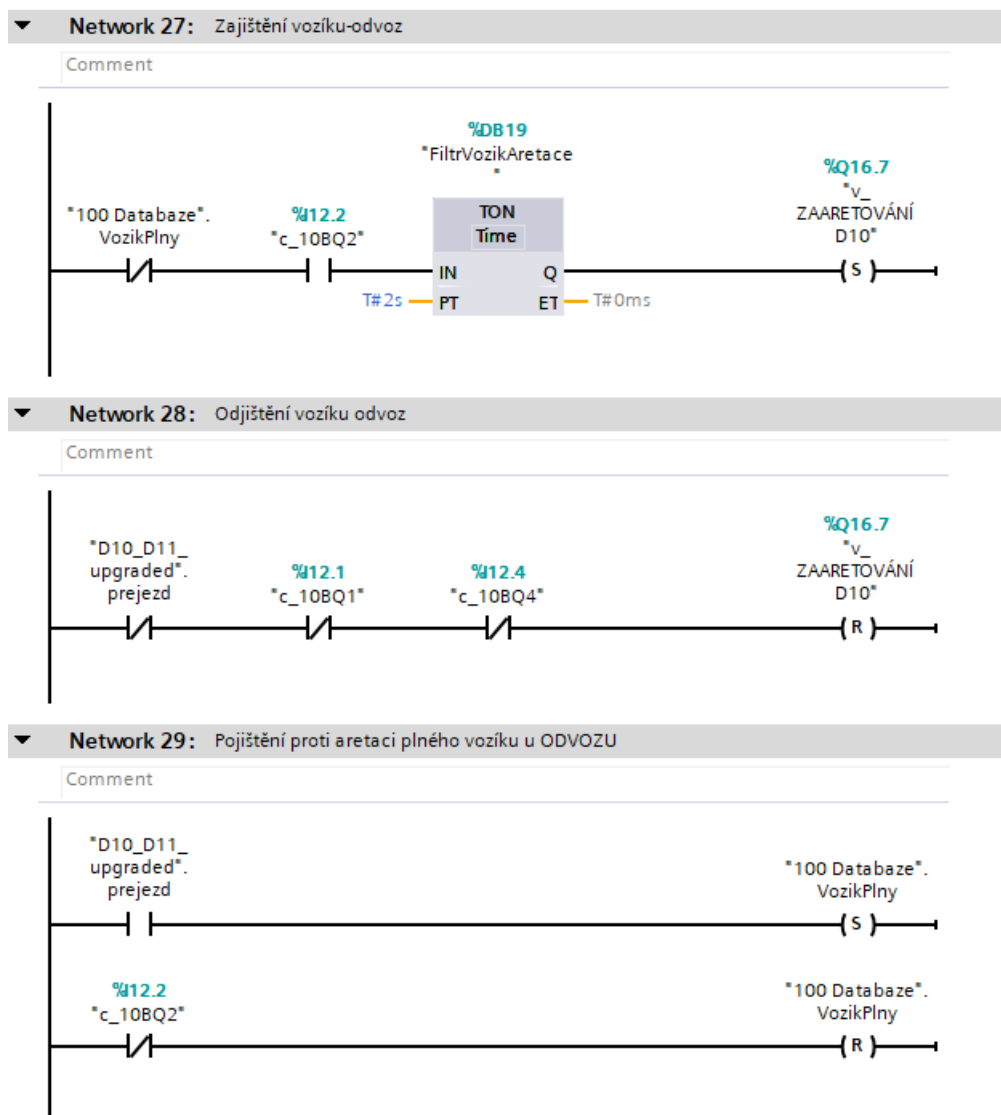
Obr. 6.12: Zajištění vozíku na začátku linky - program

Zajištění vozíku na konci linky - odvoz

Konec linky je řešen odlišně než začátek linky. V networku číslo 27 je zajištění vozíku za následujících podmínek. Vozík je přítomen (*c_10BQ2*), vozík je prázdný (*VozikPlny*) a uplynul časový interval nastavený na časovači.

Network číslo 28 má odjistit vozík v případě, že není aktivní přejezd na vozík (*.prejezd*). Dále čidlo přítomnosti palety na posledním dopravníku (*c_10BQ1*) je nezakryté a čidlo kontroly přejetí na vozík (*c_10BQ4*) je taktéž nezakryté.

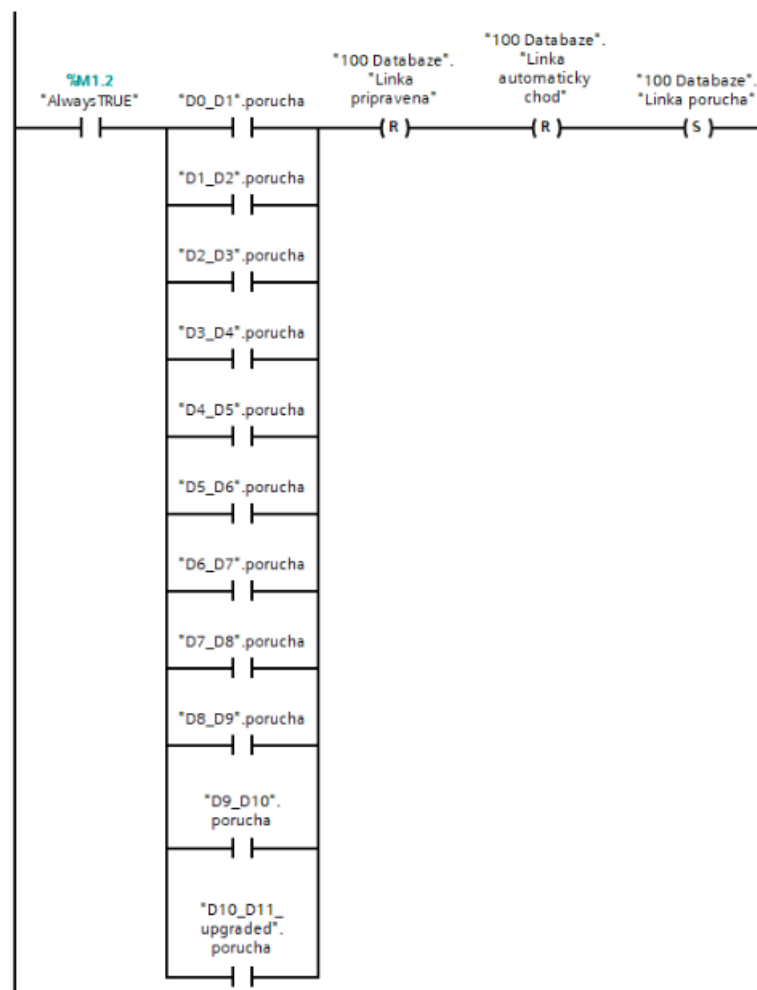
Poslední network 29 zajišťuje, aby v nedošlo k opětovnému zajištění vozíku na který byla přesunuta paleta. Pomocný bit *VozikPlny* je setován pokud je aktivní přejezd a resetován pokud je čidlo přítomnosti vozíku nezakryté.



Obr. 6.13: Zajištění vozíku na konci linky - program

Detekování poruchy u přejezdů

Kontrola zda přejezd byl vykonán řádně je uvnitř funkčního bloku. Pokud zde dojde k poruše setuje se bit *porucha*. Aby byla porucha převedena z jednotlivých instancí do globálního prostoru je použit následující kód. Pokud se tak stane resetuje se bit linka připravena (*LinkaPripravena*), linka automatický chod (*LinkaAutomatickyChod*) a setuje se bit linka v poruše (*LinkaPorucha*).



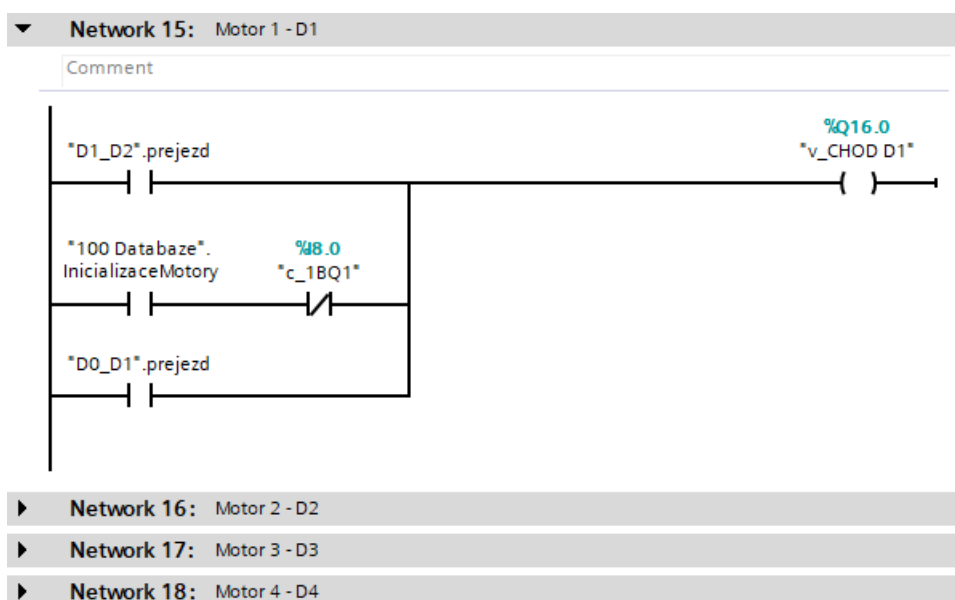
Obr. 6.14: Kontrola poruchy - program

Spínání motorů

Spínání motorů je realizováno pomocí kontaktu assignment. Každý motor má vlastní network, ty jsou popsány dle toho jaký motor/dopravník ovládají. Zde bylo lepší použít assignment kontakt než set a reset. Protože zde nejsou další podmínky pro spuštění motoru. A je přehlednější mít ovládání na jednom místě. Bylo by možné rozdělit ovládání pro přejezdy a inicializaci, ale kontakt assignment pro jednu věc je správné volat pouze jednou.

Konkrétně na obrázku 6.15 v networku 15 je ovládání motoru/dopravníku 1. K jeho sepnutí dojde pokud je aktivní přejezd mezi D1 a D2 nebo D0 a D1, tedy přesun palety z přívozního vozíku na první dopravník. Poslední cestou jak lze sepnout motor je při inicializaci, dokud není zakryté čidlo na prvním dopravníku nebo neuplyne časovač.

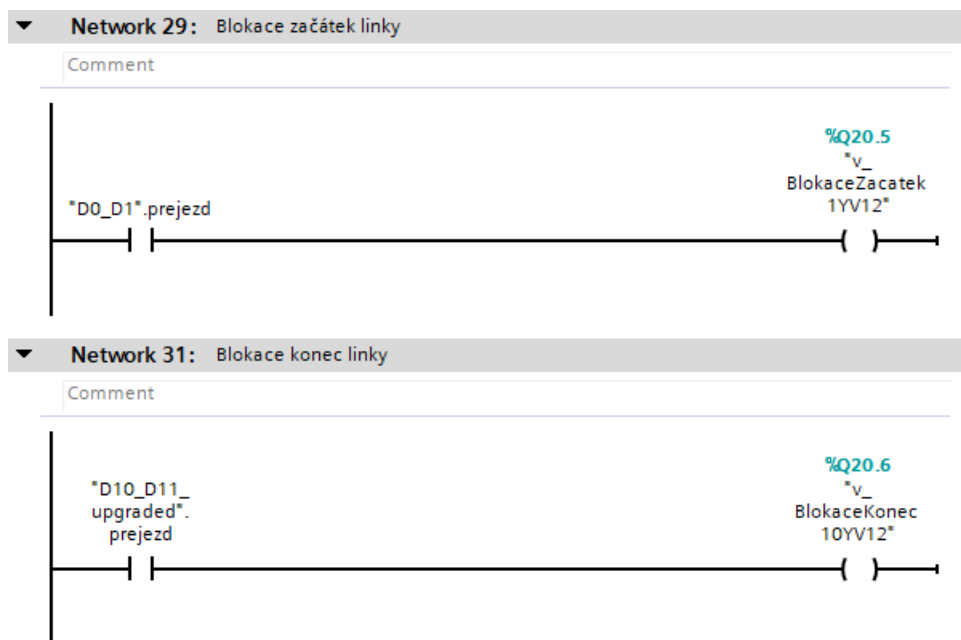
Na obrázku je ovládání motorů pouze 1 až 4, ale následují networky i pro zbytek motorů.



Obr. 6.15: Motory - program

Blokace konců linky

Blokace konců linky by bylo možné ovládat i v bloku přejezdu. Například za podmínek, že vozík je připojen lze blokaci deaktivovat. Nejjednodušší a nejjistější možností je blokaci deaktivovat až když začne probíhat přejezd, kterému by překážela. To ulehčí kontrolu kdyby něco nefungovalo tak jak má. A zamezí to pádu palety z linky. Pokud bude blokace dole a paleta spadne. Je jasné, že se stal problém s přejezdem. Nevadilo by kdyby paleta narazila do blokace to je další důvod proč je toto řešení v pořádku.



Obr. 6.16: Blokace na koncích linky - program

6.3 Modifikace dopravníkového systému

Zakázka byla zadána jak bylo popsáno v předešlých kapitolách. Ale zákazník požadoval příslib možné modifikace pokud by se jejich využívání linky mělo měnit a tedy by bylo nutné měnit i funkčnost linky. Zákazník po asi 4 měsících přišel s několika návrhy, které by pracovníkům ulehčily práci na lince. O nich je tato kapitola.

Požadavky:

- **A**-Paleta odjíždí z stanice kontroly, ale pracovník omylem odeslal paletu dříve než dokončil vše co měl. Paleta by zastavila až na stanici vyřazení. Bylo by možné aby zastavila ihned nebo co nejdříve?
- **B**-Pokud budou motory na paletu nakládány na prvním dopravníku nestane se s programem něco? Pokud budou motory z palety z posledního dopravníku skládány nespadne paleta?
- **C**-Bylo by možné, aby se na linku vešlo více palet najednou ? Jde o možnost na linku navozit více palet. A poté mít delší časovou prodlevu než bude potřeba navozit další.

Řešení:

- **A**-Paleta bude zastavena až na nejbližším dopravníku po zmáčknutí tlačítka blokovat u stanice kontroly. Paleta bude stát na daném dopravníku a dokud obsluha nezmačkne tlačítko blokovat u stanice vyřazení poté bude paleta odeslána a linka pokračuje v normální funkci.

Implementace byla vyřešena tím že se na potvrzení pro přejezd D6 a D7 se na vstup přivedl pomocný bit. Tento bit je primárně nastaven na 1, ale pokud dojde k zmáčknutí tlačítka na stanovišti kontroly resetuje se bit na hodnotu 0. Tím dojde k zastavení jakéhokoli přejezdu z D5, D6, D7. Pro jeho setování a povolení přejezdů musí pracovník zmáchnout tlačítko blokace u stanice vyřazení.

Pro názornost bylo v bloku signalizace definováno, že tlačítko blokace na stanovišti kontroly svítí pokud je možné zastavit palety. Po stlačení a aktivaci této blokace bliká a zároveň se rozsvítí tlačítko blokace na stanovišti vyřazení, protože je možné blokaci zrušit.

- **B-**Na začátek linky lze také paletu položit a bez obav o pád ji naložit. Problémem je její odeslání na další dopravník. To probíhá ihned jakmile je dopravník před prvním volný. Řešením je tedy paletu položit na kraj mimo dosah čidla a po naložení ji odsunout nad čidlo. Ano v případě, že by tato možnost byla často využívána je možné udělat odeslání palety z prvního dopravníku na další pomocí potvrzení tlačítkem. Ale zrušila by se jeho funkce na přesun palety z vozíku. Jako řešení bylo zvoleno nakládání mimo detekci čidla. Protože se jedná o výjimečné stavy a primárně se palety vozí na vozíku.
- **C-**Na lince je dostatek prostoru pro další palety. Jeden dopravník určený pro jednu paletu je asi o 60 cm delší než sama paleta. Tedy před stanicí kontroly se aktuálně mohou nashromáždit 4 palety. Prostorově by bylo možné jich nashromáždit asi 6. Problém je jak dopravovat jednotlivé palety. Protože pokud by se skládaly například 10 cm od sebe pak by se nedaly jednotlivě dopravovat. Čidlo by vidělo mezeru mezi nimi na krátký časový interval. To by považovalo za šum. Pokud by byl filtr tohoto šumu odstraněn, mohl by personál nechtěně činností zakrýt čidlo a spustit nežádoucí přejezd, ten by samozřejmě spustil poruchu.

Z pohledu programátora by bylo možné tuto situaci řešit přidavnými čidly pro přítomnost palet. A poté částečně hromadit palety za sebe. Nebylo by to ale zcela dokonalé řešení z důvodu srážek palet a jiných možných komplikací. Zde navrhované řešení přidání čidel, pozastavení provozu a příslib asi jedné nebo dvou palet navíc na lince zákazník odmítl.

Jako konstrukční řešení se nabízí rozdělit dopravníky na menší tak aby velikostně odpovídaly jedné paletě. Princip dopravy by zůstal stejný. Ale toto řešení by bylo velice nákladné a zabránilo by provozu na delší dobu. Proto ho zákazník odmítl.

7 Odladění a zhodnocení dopravníkového systému

7.1 Odladění dopravníkového systému

Programování probíhalo pouze z malé části bez dopravníkového systému, kde bylo možné vše odzkoušet. Části, které byly naprogramovány bez možnosti přístupu k dopravníkovému systému byly stav linky a signalizace. Stav linky bylo možné vyzkoušet pouze s rozvaděčem, který byl k dispozici. Protože například nouzová tlačítka nebyla připojena, nabízela se dvě řešení. Prvním řešením bylo drátem pomyslně zkratovat tyto tlačítka. A druhou možností vyměnit typ kontaktu v programu, spínací za rozpínací. Odzkoušení těchto dvou hlavních bloků bylo tedy bez problémů.

Následující bloky byly realizovány až u dopravníkového systému. Bylo možné implementované části kódu ihned odzkoušet a případně upravit. Tvorba kódu probíhala ve velké části bez problémů.

Začalo se realizací přejezdů. Prvním problémem byla nevědomost práce s časovači uvnitř funkčního bloku. Protože se přejezd volal několikrát a bylo třeba aby každý přejezd měl vlastní časovač s vlastní pamětí. Tento problém byl odhalen a po radě kolegy odstraněn.

Byla otestována funkce zda všechny palety vždy zastaví na příslušných stanicích. A zda funguje jejich odeslání tlačítkem, vyjma stanic přívozu a odvozu.

Následně došlo k implementaci přívozu a odvozu pomocí vozíků. První problém se vyskytl při zajišťování vozíků. Čidlo bylo třeba seřídít opakovaně, protože nebylo řádně dotaženo a také bylo trochu na hraně rozsahu. To protože zákazník přišel s vlastními vozíky, které mají aretační očko kousek níž než dodané vozíky.

U stanice přívozu nebyl s implementací větší problém. Naopak u stanice odvozu se naskytlo problémů hned několik. Jedním problémem byl stav, kdy paleta jede mimo dopravník aniž by zde byl vozík. Tomu bylo zamezeno pomocí přidání podmínek pro start přejezdu. To pomocí přidání vstupů do funkčního bloku přejezdu. Dalším problémem bylo seřízení čidla pro detekci blokace na vozíku. Bylo zjištěno že signál, který dostáváme od čidla je takřka náhodný. Došlo k lehčí konstrukční modifikaci.

Posledním problémem bylo zajištění vozíku ihned poté co byl odpojen. Toto bylo řešeno pomocným bitem, který se měnil až poté co čidlo přítomnosti vozíku zachytilo, že vozík někdo odtáhl. Když se vykonávala modifikace funkce. Bylo provedeno jen málo malých změn. Jedinou kterou bylo třeba odzkoušet bylo blokování převozu palety dál po lince pokud dojde k stisku tlačítka blokovat na stanici kontroly. Tato přidaná funkce byla otestována.

7.2 Zhodnocení a návrhy na zdokonalení

Celý systém byl odzkoušen a uveden v činnost u zákazníka. S odstupem asi tří měsíců od byla provedena modifikace pro zákazníka. Bylo tedy možné se za provozu přesvědčit, že vše funguje správně. Po konstrukční stránce je řízená technologie bez větších nedostatků. Po stránce řízení je také v pořádku. Z obou hledisek by bylo možné provést několik vylepšení.

Lze konstatovat, že funkčnost dopravníkového systému je dobře implementována. Protože práce na zařízení je efektivní. Pracovníci se nikde nezdržují nad funkcí či pomalostí linky. K poruchám dochází pouze výjimečně a zákazník nijakou pravidelnou poruchu nehlásil.

Pro zdokonalení dopravníkového systému byly navrženy následující změny:

- Rozdělení dopravníků na menší samostatně ovládané dopravníky, aby bylo možné nahromadit za sebe více palet na linku.
- Zlepšení konstrukčního řešení vozíků. Špatně se s nimi manipuluje pokud se snažíte najet správně a rovně ke konci dopravníků.
- Umístění tlačítka přivolat a odeslat u konců dopravníku dát na dosah ruky obsluhy, která přivezla vozík. Aby nemusela popocházet.
- Bylo by možné vytvořit program, který by v rozlišoval palety na dobré a s špatným kusem. Toto odlišení by se provedlo dle vyhodnocení pracovníka na stanici kontroly. Zmáčkl by tlačítko odeslat nebo blokovat. V případě odeslat by paleta jela až na konec linky. V druhém případě by paleta sama zastavila na stanici vyřazení, kde by čekala na ruční odeslání.

8 Závěr

Bakalářská práce se zabývá dopravníkovým systémem pro zkušebnu elektromotorů. Na základě konzultací se zadavatelem byly definovány požadavky na řízení technologie. Byly prostudovány parametry jednotlivých čidel, které vybrala dodavatelská firma, a ověřena jejich vhodnost pro danou aplikaci.

Na základě definovaných požadavků na řízení dopravníkového systému byla stanovena koncepce řídicího software pro programovatelný automat SIMATIC S7-1500 a definovány způsoby ošetření chybových stavů. Řídicí program byl rozdělen do bloků a implementován ve vývojovém prostředí Tia Portal.

Implementovanou funkcí dopravníkového systému je převoz palet po lince. Přičemž palety jsou na vozíku přiváženy a přesouvány na dopravník a na konci dopravníku jsou přesunuty na vozík a odvezeny. A poté jsou dopraveny na příslušné stanice. Program vyhodnocuje signály z 22 čidel, ovládá 10 motorů a 4 pneumatické válce.

Do rozvaděče byl nainstalován programovatelný automat a připojeny čidla, ovládací prvky a akční členy. Následně byl odladěn u zákazníka. Na základě požadavků zákazníka byly provedeny úpravy na řízení a konstrukce technologie. Pro obsluhu dopravníkového systému byl sepsán návod k obsluze a předán zákazníkovi. Dodavateli zakázky bylo slovně předáno hodnocení vytvořené technologie a návrhy na její zlepšení. To po stránce řízení i mechanické konstrukce.

Literatura

- [1] Programovatelný logický automat. *Wikipedie* [online]. [2019] [cit. 2019-12-31]. Dostupné z: https://cs.wikipedia.org/wiki/Programovateln%C3%BD_logick%C3%BD_automat
- [2] Kompaktní PLC. *TME* [online]. 2014 [cit. 2019-12-31]. Dostupné z: https://www.tme.eu/html/gfx/ramka_4535.jpg
- [3] Modulární PLC. *Automatizace.hw* [online]. 2014 [cit. 2019-12-31]. Dostupné z: <https://automatizace.hw.cz/io-moduly-s-prislusenstvim-pro-simatic-s7300>
- [4] PLC jazyky. *ELUC* [online]. Olomouc [cit. 2019-12-31]. Dostupné z: <https://eluc.kr-olomoucky.cz/verejne/lekce/976>
- [5] Základní jednotka SIMATIC S7-1200. *Technoline* [online]. [cit. 2020-01-03]. Dostupné z: <https://eshop.technoline.cz/ridici-jednotky-cpu-simatic-s7-1200/368997-zakladni-jednotka-simatic-s7-1200-cpu-1214c-dcdcdc-6es7214-1ag40-0xb0-siemens>
- [6] Řídicí reléový PLC. *Conrad* [online]. [cit. 2020-01-03]. Dostupné z: <https://www.conrad.cz/ridici-releovy-plc-modul-siemens-cpu-1214c-dc-dc-dc-6es7214-1ag31-0xb0-ip20.k197468>
- [7] Rozšiřovací PLC modul 1221. *Conrad* [online]. [cit. 2020-01-04]. Dostupné z: <https://www.conrad.cz/rozsirovaci-plc-modul-siemens-sm-1221-6es7221-1bh32-0xb0.k197708>
- [8] Rozšiřovací PLC modul 1222. *Conrad* [online]. [cit. 2020-01-04]. Dostupné z: <https://www.conrad.cz/rozsirovaci-plc-modul-siemens-sm-1222-6es7222-1bh32-0xb0.k197667>
- [9] Nový Tia portal. *Industry forum* [online]. 2017 [cit. 2020-01-04]. Dostupné z: <https://www.industryforum.cz/novy-tia-portal-vam-urychli-a-zjednodusi-praci>
- [10] Ovládací prvek. *Shop dzsc* [online]. [cit. 2020-01-05]. Dostupné z: <http://shop.dzsc.com/stk/3sb3802-0aa3.html>
- [11] Elektromotor. *Elektromotory Moravec* [online]. [cit. 2020-01-05]. Dostupné z: <https://www.elektromotory.net/siemens/1la7-1500-otacek/1le1002-0cb22-0-25kw-1365ot.html>
- [12] Signalizační sloupek. *KV Elektro* [online]. [cit. 2020-01-05]. Dostupné z: <https://www.e1.cz/produkt/1196444-signalizacni-sloupek-cervena-zluta-zelena-24vac-dc-ip66-eaton-sl7-100-l-ryg-24led-171425?t=popis>

- [13] Ultrazvukové čidlo. *Sick* [online]. [cit. 2020-01-05]. Dostupné z: <https://www.sick.com/cz/cs/snimace-mereni-vzdalenosti/ultrazvukove-snimace/um18/um18-211161101/p/p334354>
- [14] Indukční čidlo. *Balluff* [online]. [cit. 2020-01-05]. Dostupné z: <https://www.balluff.com/local/cz/productfinder/product/?key=BES05EU#/>
- [15] PLC Siemens *PLC Siemens* [online]. [cit. 2020-02-28]. Dostupné z: <https://new.siemens.com/cz/cs/products/automation/systems/industrial/plc/s7-1200.html>
- [16] *Festo-čidlo v pneuválci* [online]. [cit. 2020-04-03]. Dostupné z: <https://cz.rs-online.com/web/p/spinace-pro-pneumaticke-valce-a-ovladace/0210250/>
- [17] *Pepperl fuchs - czech republic* [online]. [cit. 2020-04-20]. Dostupné z: <https://www.conrad.cz/reflexni-opticky-snimac-pepperl-fuchs-glv18-8-h-120-73-120-m12-dosah-15-120-mm.k156487>
- [18] Pneumatický píst festo. *Landefeld* [online]. [cit. 2020-05-31]. Dostupné z: <https://www.landefeld.com/artikel/cs/adn-32-10-a-p-a-536269-compact-cyl-/OT-FESTO000085>

Seznam symbolů, veličin a zkratk

PLC	Programable logic controler
CPU	central processing unit
DI	digital input
DO	digital output
V	volt
A	ampér
DC	stejnoseměrný (direct curent)
AC	střídavý (alternating curent)